

形态学图像处理：边界提取

2019年11月1日 13:44

简介

简介没啥说的，直接看原理吧

原理

表示为 $\beta(A)$ 的集合 A 的边界可以通过先用 B 对 A 腐蚀，而后执行 A 和腐蚀的结果之间的集合之差得到，即

$$\beta(A) = A - (A \ominus B)$$

其中 B 是一个适当的结构元。

代码

```
clc;clear;
I = imread('cameraman.tif');
I = imbinarize(I, graythresh(I));
figure();imshow(I);title('cameraman二值化图像');
I = ~I;
figure();imshow(I);title('cameraman二值化图像取反');
%代码实现边界提取
[m, n] = size(I);
Ie1 = zeros([m, n]);
for i = 2:m - 1
    for j = 2:n - 1
        Ie1(i, j) = sum(I(i - 1:i + 1, j - 1:j + 1), 'all') == 9;
    end
end
Ib1 = abs(I - Ie1);
figure();imshow(Ie1);title('代码实现腐蚀结果');
figure();imshow(Ib1);title('代码实现边界提取');
%函数实现边界提取
se = strel('square', 3);
Ie2 = imerode(I, se);
Ib2 = abs(I - Ie2);
figure();imshow(Ie2);title('函数实现腐蚀结果');
figure();imshow(Ib2);title('函数实现边界提取');
```

结果分析

图像用的还是经典的cameraman（老师要求的👨‍🎓），先利用graythresh得到的阈值对图像二值化

cameraman 二值化图像



由于对二值图像的腐蚀定义为对像素值为1的部分的腐蚀，而cameraman的目标部分像素值为0，背景部分像素值为1，因此先将图片取反

cameraman 二值化图像取反



代码实现的图像腐蚀及边界提取

代码实现腐蚀结果



代码实现边界提取



利用imerode函数实现的图像腐蚀及边界提取

函数实现腐蚀结果



函数实现边界提取

